



# Einleitung

Das Thema dieses Buchs ist Reinforcement Learning (verstärkendes Lernen), ein Teilgebiet des Machine Learnings. Es konzentriert sich auf die anspruchsvolle Aufgabe, optimales Verhalten in komplexen Umgebungen zu erlernen. Der Lernvorgang wird ausschließlich durch den Wert einer Belohnung und durch Beobachtung der Umgebung gesteuert. Das Modell ist sehr allgemein und auf viele Situationen anwendbar, von einfachen Spielen bis hin zur Optimierung komplexer Fertigungsprozesse.

Aufgrund der Flexibilität und der allgemeinen Anwendbarkeit entwickelt sich das Fachgebiet Reinforcement Learning sehr schnell weiter und weckt großes Interesse bei Forschern, die vorhandene Methoden verbessern oder neue Methoden entwickeln wollen, und bei Praktikern, die ihre Aufgaben möglichst effizient bewältigen möchten.

## Motivation

Dieses Buch stellt den Versuch dar, dem Mangel an praxisnahen und strukturierten Informationen über Verfahren und Ansätze des Reinforcement Learnings (RL) entgegenzuwirken. Es gibt weltweit umfassende Forschungsaktivitäten, und fast täglich werden neue Artikel veröffentlicht. Große Teile von Deep-Learning-Konferenzen wie NeurIPS (Neural Information Processing Systems) oder ICLR (International Conference on Learning Representations) widmen sich RL-Verfahren. Es gibt mehrere große Forschungsgruppen, die sich auf die Anwendung von RL-Verfahren in der Robotik, in der Medizin und auf Multiagenten-Systemen konzentrieren.

Die Informationen über die jüngsten Forschungsergebnisse sind zwar allgemein verfügbar, sie sind aber zu spezialisiert und zu abstrakt, um sie ohne erhebliche Anstrengung zu verstehen. Bei den praktischen Aspekten von RL-Anwendungen ist die Situation noch schlimmer, weil oft nicht klar ist, wie man von der abstrakten mathematischen Beschreibung einer Methode in einem Forschungsartikel zu einer funktionierenden Implementierung gelangt, die eine Aufgabe tatsächlich löst.

Das erschwert es am Fachgebiet Interessierten, die Methoden und Konzepte, die in Artikeln oder Konferenzvorträgen vorgestellt werden, unmittelbar zu verstehen. Es gibt ganz ausgezeichnete Blogbeiträge über verschiedene RL-Methoden, die durch funktionierende Beispiele veranschaulicht werden, aber das eingeschränkte Format eines Blogbeitrags ermöglicht es dem Autor nicht, mehr als ein oder zwei Methoden zu beschreiben, ohne den vollständigen Kontext darzustellen und zu zeigen, in welcher Beziehung die verschiedenen Methoden zueinanderstehen. Dieses Buch ist mein Versuch, dieses Problem in Angriff zu nehmen.

## Der Ansatz

Ein weiterer Aspekt des Buchs ist die Praxisorientierung. Alle Methoden werden in verschiedenen Umgebungen implementiert, die von völlig trivial bis zu ziemlich komplex reichen. Ich habe versucht, die Beispiele so zu gestalten, dass sie leicht verständlich sind, was durch die Leistungsfähigkeit von PyTorch ermöglicht wurde. Die Komplexität und die Anforderungen der Beispiele orientieren sich an RL-Interessierten, die keinen Zugang zu sehr großer Rechenleistung haben, wie einem GPU-Cluster oder sehr leistungsstarken Workstations. Dadurch wird, wie ich hoffe, das hochinteressante und spannende Fachgebiet RL einem breiteren Publikum zugänglich, nicht nur Forschungsgruppen oder großen KI-Unternehmen. Allerdings geht es nach wie vor um **Deep** Reinforcement Learning, deshalb empfiehlt sich die Verwendung einer GPU. Etwa die Hälfte der Beispiele im Buch profitiert davon, wenn sie auf einer GPU ausgeführt werden.

Beim Reinforcement Learning kommen oft Umgebungen mittlerer Größe zum Einsatz, beispielsweise bei Atari-Spielen oder für kontinuierliche Steuerungsaufgaben, das Buch enthält aber auch einige Kapitel (Kapitel 10, 14, 15, 16 und 18), in denen größere Projekte beschrieben werden, um zu veranschaulichen, wie sich RL-Verfahren auf kompliziertere Umgebungen und Aufgaben anwenden lassen. Diese Beispiele sind allerdings auch keine vollständigen Projekte aus der Praxis (sonst würden sie ein eigenes Buch erfordern), sondern etwas umfassendere Aufgaben, die veranschaulichen, wie sich das RL-Paradigma auf Bereiche jenseits der üblichen Benchmarks anwenden lässt.

Bei den Beispielen in den ersten drei Teilen des Buchs habe ich versucht, den vollständigen Quellcode zu zeigen, damit die Beispiele eigenständig sind. In einigen Fällen führt das dazu, dass Teile des Codes wiederholt werden (beispielsweise sind die Trainings-Schleifen der meisten Verfahren sehr ähnlich), aber ich denke, direkt zu einer Methode von Interesse springen zu können, ist wichtiger, als einige Wiederholungen zu vermeiden. Alle Beispiele im Buch sind auf Github verfügbar (<https://github.com/PacktPublishing/Deep-Reinforcement-Learning-Hands-On-Second-Edition>) und Sie sind herzlich eingeladen, mit dem Code zu experimentieren und eigene Beiträge zu leisten.

## Für wen ist das Buch gedacht?

Das Buch richtet sich vornehmlich an alle, die schon über einige Vorkenntnisse im Bereich Machine Learning verfügen und daran interessiert sind, Reinforcement Learning in der Praxis kennenzulernen. Der Leser sollte mit Python und den Grundlagen von Deep Learning und Machine Learning vertraut sein. Kenntnisse der Statistik und Wahrscheinlichkeitsrechnung sind von Vorteil, aber nicht unbedingt erforderlich, um den Großteil des Buchs zu verstehen.

## Zum Inhalt des Buchs

*Kapitel 1, Was ist Reinforcement Learning?*, stellt eine Einführung in die grundlegenden Ideen des Reinforcement Learnings und der wichtigsten formalen Modelle dar.

*Kapitel 2, OpenAI Gym*, führt Sie anhand der Open-Source-Bibliothek Gym in die praxisnahen Aspekte des RL ein.

*Kapitel 3, Deep Learning mit PyTorch*, gibt einen Überblick über die PyTorch-Bibliothek.

*Kapitel 4, Das Kreuzentropie-Verfahren*, stellt eines der einfachsten RL-Verfahren vor, um Ihnen einen Eindruck von RL-Verfahren und RL-Aufgaben zu vermitteln.

*Kapitel 5, Tabular Learning und das Bellman'sche Optimalitätsprinzip*, führt in die wertebasierten RL-Verfahren ein.

*Kapitel 6, Deep Q-Networks*, beschreibt DQNs, die Erweiterung elementarer wertebasierter Verfahren, die es ermöglichen, Lösungen für komplexere Umgebungen zu finden.

*Kapitel 7, Allgemeine RL-Bibliotheken*, stellt die PTAN-Bibliothek vor, die wir im Buch verwenden werden, um die Implementierung von RL-Verfahren zu erleichtern.

*Kapitel 8, DQN-Erweiterungen*, gibt einen detaillierten Überblick über moderne Erweiterungen von DQNs, die zur Verbesserung der Stabilität und der Konvergenz in komplexen Umgebungen dienen.

*Kapitel 9, Beschleunigung von RL-Verfahren*, bietet eine Übersicht über die Möglichkeiten, die Ausführung von RL-Code zu beschleunigen.

*Kapitel 10, Aktienhandel per Reinforcement Learning*, erläutert das erste konkrete Projekt, die Anwendung des DQN-Verfahrens auf den Aktienhandel.

*Kapitel 11, Eine Alternative: Policy Gradients*, stellt eine weitere Familie wertebasierter RL-Verfahren vor, die auf Policy Learning beruhen.

*Kapitel 12, Das Actor-Critic-Verfahren*, beschreibt eines der am häufigsten verwendeten RL-Verfahren.

*Kapitel 13, Asynchronous Advantage Actor Critic*, erweitert das Actor-Critic-Verfahren durch parallele Kommunikation mit der Umgebung, um Stabilität und Konvergenz zu verbessern.

*Kapitel 14, Chatbot-Training per Reinforcement Learning*, beschreibt das zweite konkrete Projekt und zeigt, wie RL-Verfahren auf NLP-Aufgaben angewendet werden.

*Kapitel 15, Die TextWorld-Umgebung*, stellt die Anwendung von RL-Verfahren auf Spiele des Genres *Interactive Fiction* (IF) vor.

*Kapitel 16, Navigation im Web*, beschreibt ein weiteres größeres Projekt, nämlich die Anwendung von RL auf die Navigation im Web anhand von MiniWoB-Aufgaben.

*Kapitel 17, Stetige Aktionsräume*, erläutert die Eigenheiten von Umgebungen, die stetige Aktionsräume verwenden, und stellt weitere Verfahren vor.

*Kapitel 18, RL in der Robotik*, befasst sich mit der Anwendung von RL-Verfahren auf Aufgaben aus dem Gebiet der Robotik. In diesem Kapitel beschreibe ich die Entwicklung und das Training eines kleinen Hardware-Roboters mithilfe von RL-Verfahren.

*Kapitel 19, Trust Regions – PPO, TRPO, ACKTR und SAC*, ist ein weiteres Kapitel über stetige Aktionsräume und beschreibt die Trust-Region-Verfahren.

*Kapitel 20, Blackbox-Optimierung beim Reinforcement Learning*, beschreibt Optimierungsverfahren, die keine Gradienten in expliziter Form verwenden.

*Kapitel 21, Fortgeschrittene Exploration*, erörtert verschiedene Ansätze zur besseren Erkundung der Umgebung.

*Kapitel 22, Jenseits modellfreier Verfahren – Imagination*, stellt unter Berücksichtigung jüngster Forschungsergebnisse modellbasierte Ansätze für RL vor.

*Kapitel 23, AlphaGo Zero*, erläutert die Anwendung des AlphaGo-Zero-Verfahrens auf das Spiel »Vier gewinnt«.

*Kapitel 24, RL und diskrete Optimierung*, beschreibt die Anwendung von RL-Verfahren auf diskrete Optimierungen anhand einer Zauberwürfel-Umgebung (Rubik's Cube).

*Kapitel 25, RL mit mehreren Agenten*, stellt eine relative neue Entwicklungsrichtung bei RL-Verfahren für Situationen vor, in denen mehrere Agenten vorhanden sind.

## Verwendung des Buchs

Alle Kapitel des Buchs, die RL-Verfahren beschreiben, sind identisch aufgebaut: Zunächst werden die Motivation für das Verfahren, die theoretischen Grundlagen und die zugrunde liegenden Ideen erläutert. Anschließend betrachten wir verschiedene Anwendungen des Verfahrens auf unterschiedliche Umgebungen anhand des vollständigen Quellcodes.

Sie können das Buch also auf verschiedene Weise verwenden:

1. Um sich einen schnellen Überblick über ein Verfahren zu verschaffen, können Sie den einführenden Teil des entsprechenden Kapitels lesen.
2. Um ein besseres Verständnis der Implementierung eines Verfahrens zu erlangen, können Sie sich den Quellcode ansehen und die dazugehörigen Kommentare lesen.
3. Um ein tiefer gehendes Verständnis eines Verfahrens zu erlangen, sollten Sie versuchen, es selbst zu implementieren und zum Laufen zu bringen (meiner Ansicht nach die beste Lernmethode). Dabei können Sie den vorhandenen Quellcode als Ausgangspunkt nutzen.

Ich kann mir nur wünschen, dass Ihnen das Buch von Nutzen sein wird!

## Codebeispiele herunterladen

Die Codebeispiele aus diesem Buch können Sie unter <http://www.mitp.de/0036> herunterladen.

## Farbige Abbildungen herunterladen

Eine farbige Version der Screenshots und Diagramme in diesem Buch finden Sie ebenfalls unter <http://www.mitp.de/0036> zum Download.

## Konventionen im Buch

In diesem Buch werden verschiedene Textformatierungen verwendet, um zwischen Informationen unterschiedlicher Art zu unterscheiden. Nachstehend finden Sie einige Beispiele und deren Bedeutungen.

Schlüsselwörter, Datenbanktabellen-, Twitter-, Datei-, Ordner-, Datei- und Pfadnamen sowie URLs und Usereingaben werden im Fließtext in nicht proportionaler Schrift darge-

stellt. Zum Beispiel: »Mounten Sie die heruntergeladene Datei `WebStorm-10*.dmg` als weiteres Laufwerk Ihres Systems.«

Codeblöcke werden wie folgt dargestellt:

```
def grads_func(proc_name, net, device, train_queue):
    envs = [make_env() for _ in range(NUM_ENVS)]

    agent = ptan.agent.PolicyAgent(
        lambda x: net(x)[0], device=device,
        apply_softmax=True)

    exp_source = ptan.experience.ExperienceSourceFirstLast(
        envs, agent, gamma=GAMMA, steps_count=REWARD_STEPS)

    batch = []
    frame_idx = 0
    writer = SummaryWriter(comment=proc_name)
```

Eingaben oder Ausgaben auf der Kommandozeile sehen so aus:

```
r1_book_samples/Chapter11$ ./02_a3c_grad.py --cuda -n final
```

**Neue Ausdrücke** und **wichtige Begriffe** werden **fett** gedruckt. Auf dem Bildschirm auswählbare oder anklickbare Bezeichnungen, wie z. B. Menüpunkte oder Schaltflächen, werden in der Schriftart Kapitälchen gedruckt: »Nach einem Klick auf die Schaltfläche **ABBRECHEN** in der unteren rechten Ecke wird der Vorgang abgebrochen.«